

Dr. Queue con Blender
Una granja de render GNU
Por Alexandre Silva Costa anakinpendragon@yahoo.com.br

Este artículo tiene la intención de enseñar a usar el soft de animaciones blender con la granja de render DRQUEUE.

Granja de Render es un concepto muy próximo al Cluster, en un cluster las aplicaciones tienen sus procesos subdivididos por las máquinas de una red, para que cada una haga una parte y entregue el resultado final, haciendo el proceso de aplicaciones complejas mucho más rápido. En el caso de una Granja de Render no dividimos los procesos de una aplicación pero si los frames de una animación. Resultados prácticos: Si usted tuviera una máquina que renderiza una animación en 2 horas, si tuviera otra idéntica haría el mismo trabajo en aproximadamente una hora, si tuviera 8 máquinas el servicio estaría listo en 7.5 minutos etc.

Para el que no lo sepa explicaré lo que es el render. Cuando modelamos en un soft 3D usamos sistemas vectoriales para crear nuestros modelos, igual que hacemos en programas más simples como Corel Draw ó Inkscape, en programas 3D usamos sistemas más complejos como opengl que consiguen generar vectores 3D para generar nuestros modelos, este tipo de técnica necesita de un procesador y tarjeta gráfica razonable para poder crear modelos complejos como seres humanos animales vehículos edificios etc. Cuando tenemos reflejos sombras y brillos la cosa se complica, texturas de alta resolución.... Pocos supercomputadores consiguen crear animación compleja en tiempo real, imagínese un supercomputador que pueda reproducir Shrek en tiempo real? Habría que comprar uno para ver la película, por eso se renderiza, generamos archivos bitmap que pueden ser creadas lentamente por un ordenador normal y puestas en orden para volcarlos a archivos de video cintas y otros formatos.

Bien entonces que hace el Dr, Queue? Se aprovecha de la propiedad de los programas como blender de usar más allá de sus interfaces gráficas para renderizar imágenes y animaciones, usarán también la línea de comandos. Por ejemplo si escribimos en línea de comando:

```
#blender -b animacion.blend -s 1
```

El comando "blender -b" indica que va a renderizar un archivo, en este caso una animación blend, será renderizado solo el frame 1 como especifica el parámetro -s. Para concluir el archivo que será generado se especifica dentro del archivo animacion.blend.se creará en la carpeta que nosotros le indiquemos en el propio archivo

El drqueue gestiona la colaboración entre las máquinas esclavas, El ordenador Master corriendo el drqueue envía para los esclavos un comando similar al comando de arriba, dentro de un script, además verifica cual es el frame que hay que renderizar y a que ordenador se le envía y continúa enviando tareas hasta que finaliza el proceso de render.

Actualmente el Drqueue también funciona con Maya, Lightwave, Mental Ray, Blender, BMRT, 3Delight, Pixie, Aqsis, After Effects, Shake, Nuke and Terragen además nada impide que sea adaptado a otros programas. Tiene versiones precompiladas para Debian, Windows y Mac OS x

Instalación del DRQUEUE y BLENDER:

La instalación del blender es sencilla, solo hay que descomprimir el archivo que se baja de blender3d.org y listo. Por cuestión de compatibilidad el ejecutable debe de ser copiado o enlazado en /usr/bin y también la carpeta .blend debe de ser copiada o enlazada en el path de forma que el comando blend funcione perfectamente desde la shell de unix o la línea de comandos de windows Esto vale para los otros programas también: Por ejemplo yo enlacé el programa shake de forma que puede ser ejecutado escribiendo shake en el shell arrastrando el ejecutable y las librerías a /usr/bin con la tecla Alt pulsada y diciendole que enlazara los archivos.

El drqueue se baja de drqueue.org para la instalación de drqueue es necesario verificar la existencia de las siguientes dependencias:

```
gcc
g++
make
libgtk (2.0 o mayor)
libgtk-devel (2.0 o mayor)
tssh
```

Es necesario setear dos variables de entorno, recomiendo que se coloquen las mismas en /etc/bashrc para que una vez instalado estas variables de entorno queden seteadas para el futuro.

```
#export DRQUEUE_ROOT="/mnt/shared/drqueue"
```

```
/* Esta variable indica la carpeta de instalación, la carpeta es un ejemplo, recomendable */
```

```
#export DRQUEUE_MASTER="192.168.0.1"
```

```
/* Esta variable indica la IP de la máquina maestra */
```

También es necesario crear un usuario drqueue que será usado por cuestiones de derechos de acceso, sino el script de instalación no irá a compilar los archivos, Entonces teclee como root o con sudo o fakeroot:

```
#useradd drqueue /*no es necesario incluir una contraseña*/
```

La carpeta donde se instala el drqueue debe de estar compartida y montada en todas las máquinas con el mismo path así todas las máquinas tendrán una carpeta /mnt/shared/drqueue que apuntarán para la misma carpeta que será la maquina maestra a donde serán enviados los frames renderizados. Recomiendo usar nfs para compartir porque es natvo de linux, si no sabe hacerlo vaya a google hay varios tutoriales

Con todo lo necesario seteado basta con teclear como usuario root:

```
#make
#make INSTROOT=$DRQUEUE_ROOT install
```

Ya está todo instalado, pero es importante dar permisos de lectura escritura y ejecución a la carpeta compartida para el usuario que quiera ejecutar el drqueue, si no no podrá ejecutar ninguna de las aplicaciones que necesitan escribir en las carpetas temporales en donde esta instalado el DRQUEUE

USO:

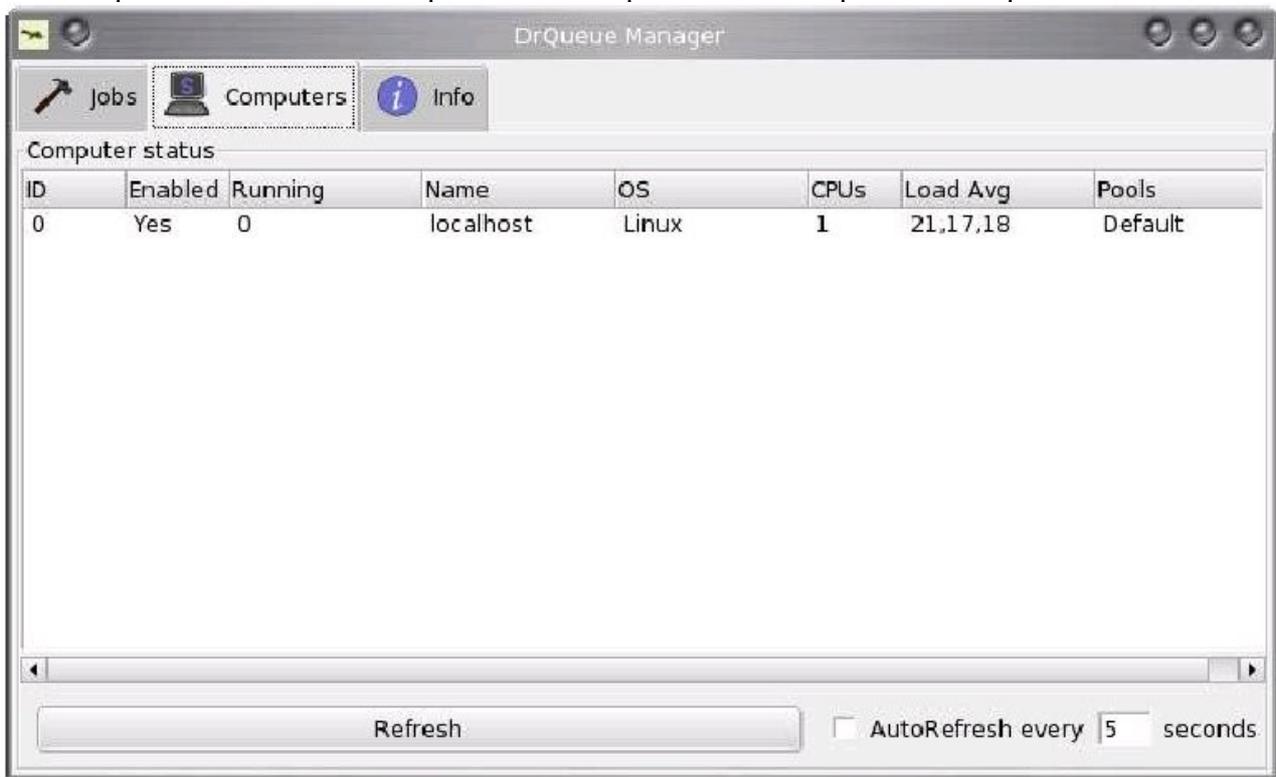
Para iniciar el master ejecutamos /mnt/shared/drqueue/bin/master

Para iniciar el esclavo ejecutamos /mnt/shared/drqueue/bin/slave

Para enviar tareas ejecutamos el drqueue manager: /mnt/shared/drqueue/bin/drqman



Para comprobar si detecta las máquinas esclavas podemos ir a la pestaña Computers:

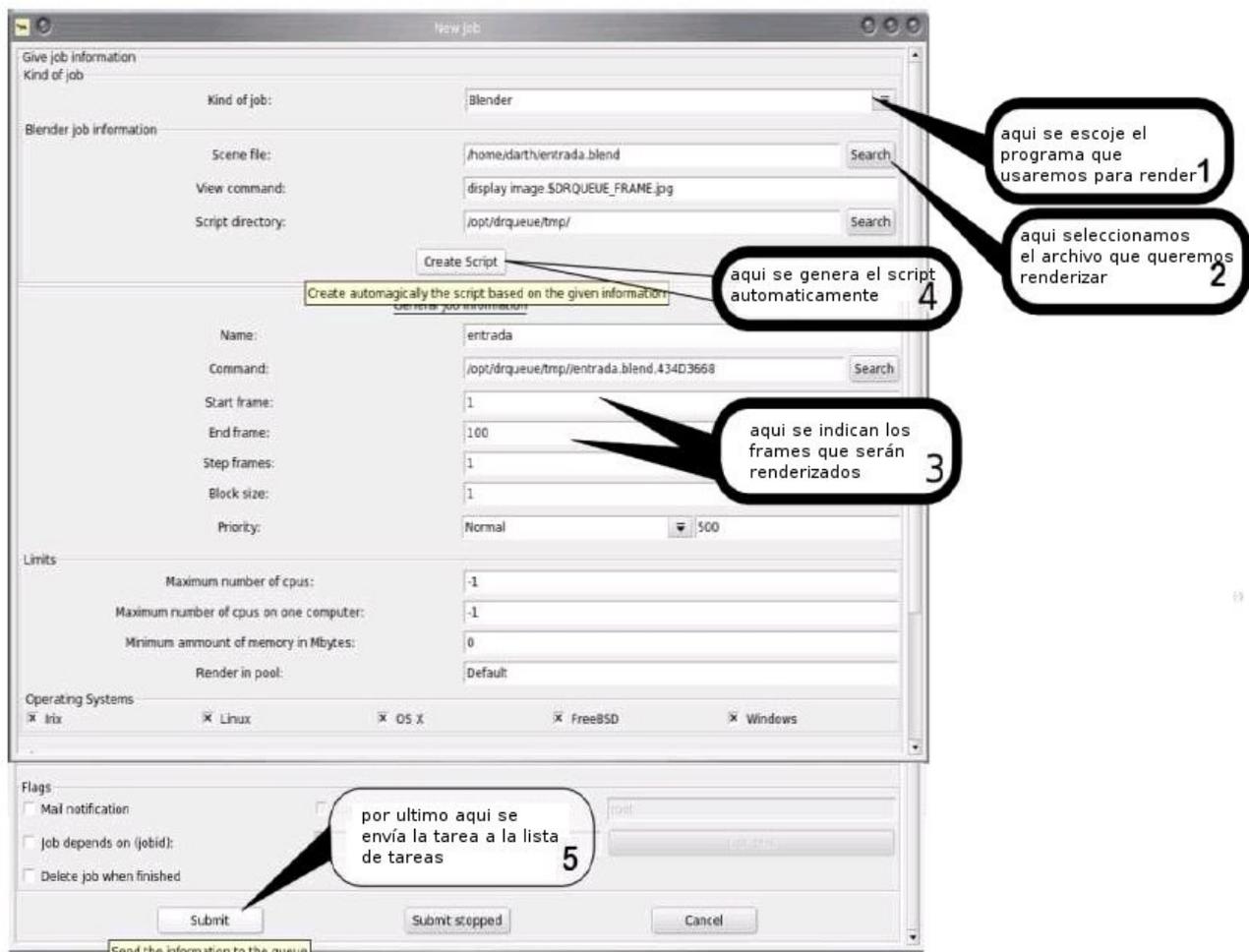


En esta página esta la lista de todos los pc esclavos

Para iniciar una nueva tarea vaya a la pestaña Jobs y haga click con el botón derecho en New Job, verá que con el botón derecho también son posibles otras acciones como cancelar ó borrar una tarea.



Despues aparece una nueva pantalla donde habrá que hacer las settings para que el archivo se renderice



1. seleccione la aplicación de origen que será utilizada para el render, en este caso el blender en “kind of job”
2. seleccione el archivo que se renderizará en “scene file”
3. En start y end frame seleccione los frames que serán renderizados
4. pulsa create script y la aplicación generará automáticamente el script que hara el servicio
5. asignele un nombre a la tarea
6. pulse submit y la tarea será enviada a la cola de render

Después de eso podremos monitorizar la marcha de la tarea en el panel principal, pero debe para eso marcar la casilla de refresco automático cada 5 segundos, sino habría que hacer el refresco manualmente



Cuando la tarea haya terminado todos los frames estarán renderizados y no habrá frames en “left”

Observaciones sobre Blender: El lugar donde serán renderizados los frames se indica en el archivo .blend generado en el blender, y no en el drqueue, por eso debemos colocar el path en el archivo para el directorio compartido en la máquina master, donde todas las máquinas deben de tener permiso de escritura. También el archivo que sera renderizado debe de estar en la misma carpeta compartida, pues el script busca en cada una de las máquinas el mismo path para el archivo que sea renderizado localmente. O sea que si los archivos en los que va a trabajar no están en esa misma carpeta compartida cuando no dea error en el render los archivos estarán desorganizados.



Como en el ejemplo de arriba, en la pestaña renderizado, en la pestaña Output, el primer path es el del archivo que será renderizado, que será en la carpeta compartida, en caso del drqueue el renderizará los archivos en secuencia, esta secuencia de archivos será transformada en video por el secuenciador del blender ó por cualquier otro editor de vídeo.

Resumiendo: El archivo que queremos renderizar “película.blend” debe de tener el mismo path en TODAS las computadoras conectadas a la red, asimismo la carpeta de Render debe de ser la misma para todas las computadoras y por último las computadoras esclavas deben de tener permiso de escritura en la carpeta compartida.

Conclusión: Blender es un soft muy interesante no solo por su coste sino también por su calidad, su atractivo aumenta además con el DRQUEUE el coste de generar una animación puede ser reducido considerablemente disminuyendo el tiempo de entrega del mismo.

Además de la satisfacción que proporciona a un cliente tener su producto listo en menos tiempo lo que aumenta nuestra competitividad. Con software propietario estas características nos costarían una fortuna.

Por último un consejo: Es posible que no tengas máquinas con linux en tu red pero si que es posible ejecutar una cola de render con un Live CD tipo Knoppix y con una llave USB donde iría instalado el DRQUEUE y el BLENDER o cualquier otro programa que quieras con lo que construirías una granja de render utilizando equipos con otros SO lo que te permite aumentar tus posibilidades de trabajo con un coste de menos de 20\$ por PC que es lo que cuesta una memoria USB de 128 MB mas que suficiente.